### **1. File IoHwAb\_SpeedSensor.h**

| #ifndef IOHWAB\_SPEEDSENSOR\_H #define IOHWAB\_SPEEDSENSOR\_H  #include "Std\_Types.h"  // Cấu hình cho cảm biến tốc độ typedef struct {  uint8\_t SpeedSensor\_Channel; // Kênh ADC để đọc giá trị từ cảm biến tốc độ  uint16\_t SpeedSensor\_MaxValue; // Giá trị tốc độ tối đa mà cảm biến có thể đọc (km/h) } SpeedSensor\_ConfigType;  // Prototype cho hàm khởi tạo cảm biến tốc độ Std\_ReturnType IoHwAb\_SpeedSensor\_Init(const SpeedSensor\_ConfigType\* ConfigPtr);  // Prototype cho hàm đọc giá trị từ cảm biến tốc độ Std\_ReturnType IoHwAb\_SpeedSensor\_Read(float\* SpeedValue);  #endif /\* IOHWAB\_SPEEDSENSOR\_H \*/ |
| --- |

### **2. File IoHwAb\_SpeedSensor.c**

| #include "IoHwAb\_SpeedSensor.h" #include "Adc.h" // Gọi API từ MCAL để đọc giá trị từ ADC #include <stdio.h> #include <stdlib.h>  // Giả lập cấu hình của cảm biến tốc độ static SpeedSensor\_ConfigType SpeedSensor\_CurrentConfig;  // Hàm khởi tạo cảm biến tốc độ với cấu hình Std\_ReturnType IoHwAb\_SpeedSensor\_Init(const SpeedSensor\_ConfigType\* ConfigPtr) {  if (ConfigPtr == NULL) {  printf("Error: Null configuration pointer passed to IoHwAb\_SpeedSensor\_Init.\n");  return E\_NOT\_OK;  }   // Lưu cấu hình cảm biến tốc độ vào biến toàn cục  SpeedSensor\_CurrentConfig.SpeedSensor\_Channel = ConfigPtr->SpeedSensor\_Channel;  SpeedSensor\_CurrentConfig.SpeedSensor\_MaxValue = ConfigPtr->SpeedSensor\_MaxValue;   // Gọi API từ MCAL để khởi tạo ADC  Adc\_ConfigType adcConfig;  adcConfig.Adc\_Channel = ConfigPtr->SpeedSensor\_Channel;  Adc\_Init(&adcConfig);   // In ra thông tin cấu hình cảm biến tốc độ  printf("Speed Sensor Initialized with Configuration:\n");  printf(" - ADC Channel: %d\n", SpeedSensor\_CurrentConfig.SpeedSensor\_Channel);  printf(" - Max Speed Value: %d km/h\n", SpeedSensor\_CurrentConfig.SpeedSensor\_MaxValue);   return E\_OK; }  // Hàm đọc giá trị từ cảm biến tốc độ Std\_ReturnType IoHwAb\_SpeedSensor\_Read(float\* SpeedValue) {  if (SpeedValue == NULL) {  return E\_NOT\_OK; // Kiểm tra con trỏ NULL  }   // Đọc giá trị từ kênh ADC  uint16\_t adcValue = 0;  if (Adc\_ReadChannel(SpeedSensor\_CurrentConfig.SpeedSensor\_Channel, &adcValue) != E\_OK) {  printf("Error: Failed to read ADC value.\n");  return E\_NOT\_OK;  }   // Chuyển đổi giá trị ADC sang tốc độ (giả lập)  \*SpeedValue = ((float)adcValue / 1023.0f) \* SpeedSensor\_CurrentConfig.SpeedSensor\_MaxValue;   // In ra giá trị tốc độ  printf("Reading Speed Sensor (ADC Channel %d): Speed = %.2f km/h\n",  SpeedSensor\_CurrentConfig.SpeedSensor\_Channel, \*SpeedValue);   return E\_OK; } |
| --- |

### **3. File Adc.h (MCAL Layer)**

| #ifndef ADC\_H #define ADC\_H  #include "Std\_Types.h"  // Cấu trúc cấu hình cho ADC typedef struct {  uint8\_t Adc\_Channel; // Kênh ADC } Adc\_ConfigType;  // Prototype cho hàm khởi tạo ADC void Adc\_Init(const Adc\_ConfigType\* ConfigPtr);  // Prototype cho hàm đọc giá trị từ kênh ADC Std\_ReturnType Adc\_ReadChannel(uint8\_t Channel, uint16\_t\* ValuePtr);  #endif /\* ADC\_H \*/ |
| --- |

### **4. File Adc.c (MCAL Layer)**

| #include "Adc.h" #include <stdio.h> #include <stdlib.h>  // Giả lập khởi tạo ADC void Adc\_Init(const Adc\_ConfigType\* ConfigPtr) {  printf("ADC Initialized for Channel %d\n", ConfigPtr->Adc\_Channel); }  // Giả lập đọc giá trị từ kênh ADC Std\_ReturnType Adc\_ReadChannel(uint8\_t Channel, uint16\_t\* ValuePtr) {  // Giả lập giá trị ADC ngẫu nhiên từ 0 đến 1023  \*ValuePtr = rand() % 1024;  printf("ADC Channel %d Read: Value = %d\n", Channel, \*ValuePtr);  return E\_OK; } |
| --- |

### **5. Ví dụ Cấu hình và Sử dụng trong Hệ thống**

#### **File main.c**

| #include "IoHwAb\_SpeedSensor.h" #include <stdio.h>  int main(void) {  // Cấu hình cảm biến tốc độ  SpeedSensor\_ConfigType speedSensorConfig = {  .SpeedSensor\_Channel = 1, // Kênh ADC 1  .SpeedSensor\_MaxValue = 200 // Tốc độ tối đa là 200 km/h  };   // Khởi tạo cảm biến tốc độ  if (IoHwAb\_SpeedSensor\_Init(&speedSensorConfig) != E\_OK) {  printf("Failed to initialize Speed Sensor.\n");  return -1;  }   // Đọc giá trị tốc độ  float speedValue = 0;  if (IoHwAb\_SpeedSensor\_Read(&speedValue) == E\_OK) {  printf("Current Speed: %.2f km/h\n", speedValue);  } else {  printf("Failed to read Speed Sensor.\n");  }   return 0; } |
| --- |

### **6. Giải thích Luồng Hoạt Động**

1. **Khởi tạo cảm biến tốc độ**:
   * Hàm IoHwAb\_SpeedSensor\_Init được gọi để khởi tạo cảm biến tốc độ với cấu hình cụ thể. Cấu hình này bao gồm kênh ADC được sử dụng để đọc giá trị từ cảm biến và giá trị tốc độ tối đa mà cảm biến có thể đo được.
   * **API Adc\_Init** từ MCAL được gọi để khởi tạo ADC cho kênh tương ứng.
2. **Đọc giá trị tốc độ**:
   * Khi cần đọc giá trị tốc độ, hàm IoHwAb\_SpeedSensor\_Read được gọi. Hàm này sử dụng **API Adc\_ReadChannel** để đọc giá trị ADC từ kênh đã cấu hình.
   * Giá trị ADC được chuyển đổi thành giá trị tốc độ dựa trên giá trị tốc độ tối đa được cấu hình.
3. **Kết quả**:
   * Giá trị tốc độ sẽ được in ra màn hình console, cho phép người dùng biết được tốc độ hiện tại mà cảm biến đo được.

### **7. Kết quả Mô phỏng**

Khi chạy chương trình trên một hệ thống giả lập, đầu ra có thể như sau:

| ADC Initialized for Channel 1 Speed Sensor Initialized with Configuration:  - ADC Channel: 1  - Max Speed Value: 200 km/h ADC Channel 1 Read: Value = 845 Reading Speed Sensor (ADC Channel 1): Speed = 165.29 km/h Current Speed: 165.29 km/h |
| --- |